7
Conclusões

De acordo com os experimentos realizados com o repositório WEBBASE na Seção 4.5, podemos observar que a restrição de *politeness* provoca grande impacto no *freshness* deste repositório. O repositório WEBBASE chega a perder quase 20% de *freshness* quando inserimos a restrição de *politeness*, em comparação com a solução ótima proposta em (Cho03a) para o problema sem esta restrição. Entretanto, conforme discutido na Capítulo 1 a restrição de *politeness* não pode ser ignorada, pois o *crawler* pode ser bloqueado por servidores *Web* caso não respeite esta restrição.

A restrição de *politeness* é apresentada na Definição 1.2 em função dos instantes de requisições aos servidores *Web*, porém modelos para este problema que levam em conta estes instantes podem ser ineficientes devido à taxa elevada de revisitações realizadas pelos *crawlers* de máquinas de busca. Duas medidas são tomadas nesta tese para a construção de políticas de revisitação eficientes: (i) fixamos *a priori* alguma política de tempo de implementação eficiente que dependa apenas das frequências de revisitações das páginas, e (ii) propomos um conjunto de restrições sobre estas frequências de revisitações que devem ser respeitadas por todas as políticas que respeitam a restrição de *politeness*. Com a medida (ii) podemos inserir no modelo algum impacto da restrição de *politeness*, sem a necessidade de inserir os instantes de requisições aos servidores. Devido ao porte do problema, consideramos que uma política é eficiente se o tempo médio para escalonar uma revisitação é sublinear no número de páginas do repositório.

Nesta tese avaliamos políticas eficientes de revisitação de páginas *Web*. Fixamos duas políticas de tempo: (i) igualmente espaçada por página e (ii) igualmente espaçada por servidor. Como a política de tempo igualmente espaçada por página pode violar a restrição de *politeness*, investigamos uma variação simples desta política chamada DELAYED. A restrição de *politeness* é facilmente respeitada na política de tempo igualmente espaçada por servidor, mas é necessário estabelecer uma regra para seleção de páginas em cada requisição ao servidor. Duas destas regras são investigadas, chamadas MERGE e RANDOM. As políticas DELAYED, MERGE e RANDOM são avaliadas

experimentalmente através da simulação do uso destas políticas para manter o nível de atualização do repositório WEBBASE. Podemos concluir que o *freshness* do repositório fornecido pelas políticas DELAYED e MERGE é praticamente o mesmo, e muito próximo ao melhor *freshness* obtido por política que respeitam a restrição de *politeness*. A política RANDOM perde cerca de 6% de *freshness* do repositório em comparação com as políticas DELAYED e MERGE. Provamos fatores de aproximação para o *freshness* do repositório quando aplicamos as política MERGE ou RANDOM. Demonstramos que 0,77 é um limite inferior para o fator de aproximação da política RANDOM, e apresentamos uma conjectura de que 0,927 é um limite inferior para o fator de aproximação da política MERGE.

A principal contribuição desta tese é mostrar que algumas políticas de revisitação simples e eficientes perdem pouco em termos do *freshness* do repositório, mesmo quando é necessário respeitar a restrição de *politeness*. Portanto, temos pouco ganho em empregar políticas mais sofisticadas e menos eficientes. Mais detalhes sobre as contribuições são apresentados na Seção 1.3. Fornecemos a seguir algumas direções para trabalhos futuros.

7.1 Trabalhos Futuros

- Fornecemos no Corolário 5.8 uma expressão para o fator de aproximação do *freshness* do repositório quando aplicamos a política MERGE. Uma avaliação numérica desta expressão permitiu formular a Conjectura 5.9, afirmando que este fator de aproximação é maior que 0,927. Embora este limite inferior para o fator de aproximação tenha sido observado nas simulações, ainda é necessário realizar uma prova formal deste resultado.
- Durante toda a pesquisa utilizamos o *freshness* como medida de atualização do repositório. É importante avaliar as políticas considerando também outras métricas. Como ponto de partida sugerimos provar fatores de aproximação para o *age* do repositório, descrito na Seção 2.1.2. Algumas métricas consideram mais detalhes ao determinar o impacto da desatualização das páginas sobre as consultas às máquinas de busca, como por exemplo as métricas *embarrassment* (Wol02) e *longevity* (Ols08). Estes detalhes podem dificultar a determinação dos fatores de aproximação, devido ao acréscimo de variáveis de entrada.
- Não consideramos o tempo de *download* das páginas, visto que na prática este tempo é em média muito inferior ao tempo mínimo permitido entre requisições consecutivas a um servidor. Entretanto, a avaliação das políticas é mais fiel à realidade quando estes tempos são considerados.

Neste caso, o modelo deve levar em conta os tamanhos da páginas e a taxa média de comunicação com cada servidor, que pode flutuar com o tempo. Quando consideramos os tempos de *download*, podemos avaliar também políticas que utilizam este tempos na construção do escalonamento de revisitações, como é o caso do *crawler* Mercator (Hey99). Os tempos de *download* também tornam mais realista, embora mais complexa, a avaliação teórica da chance de congestionamento do canal de comunicação realizada na Seção 5.4.

– As importâncias das páginas não impactam os fatores de aproximação para o *freshness* do repositório, mas podem afetar os resultados experimentais caso exista relação entre taxa de modificação e importância das páginas, ou relação entre número de páginas no servidor e importância das páginas. Se desejamos maior peso no cálculo do *freshness* do repositório para as páginas com maior chance de serem retornadas pela máquina de busca, então podemos utilizar por exemplo o *page rank* (Pag98) como importância da página. O *log* de consultas também pode ser utilizado para determinar a chance da máquina de busca retornar cada página. Temos então uma simulação mais realista quando consideramos as importâncias das páginas. Podemos observar se estas importâncias impactam de forma significativa o *freshness* do repositório, embora este impacto esteja limitado pelos fatores de aproximação.

– Investigar a redução do impacto da restrição de *politeness* quando exploramos o fato de algumas páginas estarem replicadas em mais de um servidor.